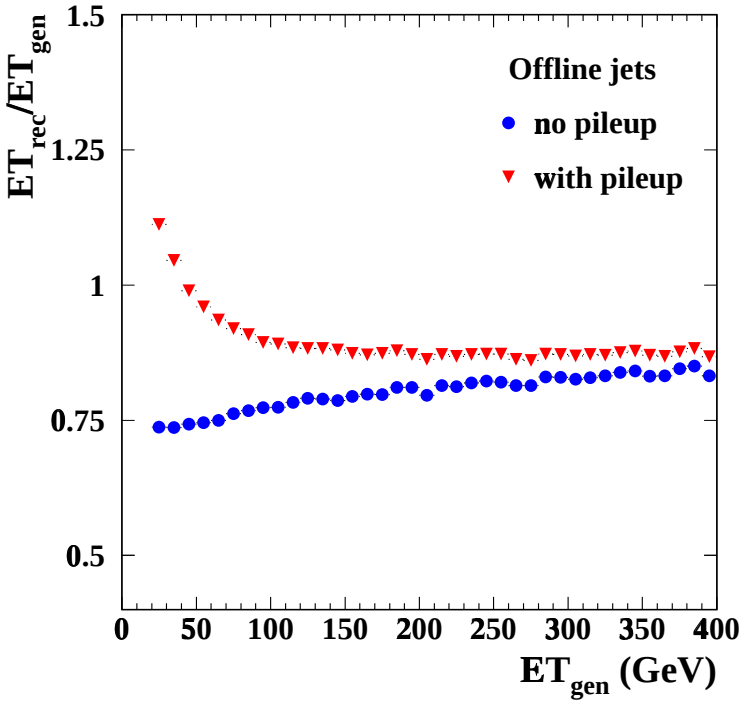
STATUS of HCAL in ORCA

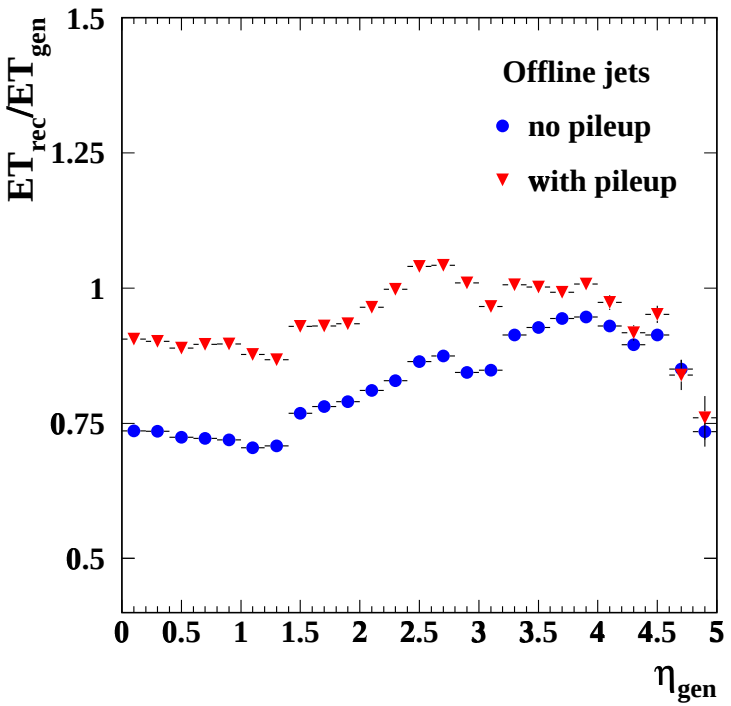
S.Abdullin, UMD

- **Jet energy scale**
- **Jet energy corrections (in ORCA)**
- **Jets/MET Ntuple maker**
- **Towards high-level (persistent) objects in ORCA**
  - **MET**
  - **τ**
  - **jet**
- **Update on HCAL electronics**

JET ENERGY SCALE

- ECAL + HCAL system has non-linear response (hadrons, jets)
- HCAL is calibrated with E_T = 50 GeV single pions
  - non-interacting in ECAL
  - too hard for low-energy jets
- Jets from ORCA 4_2_0 (+ CMSIM 116)
  - IterativeCone agorithm
  - R = 0.5
  - "gen" : stable particles
  - "rec" : EcalPlusHcalTower

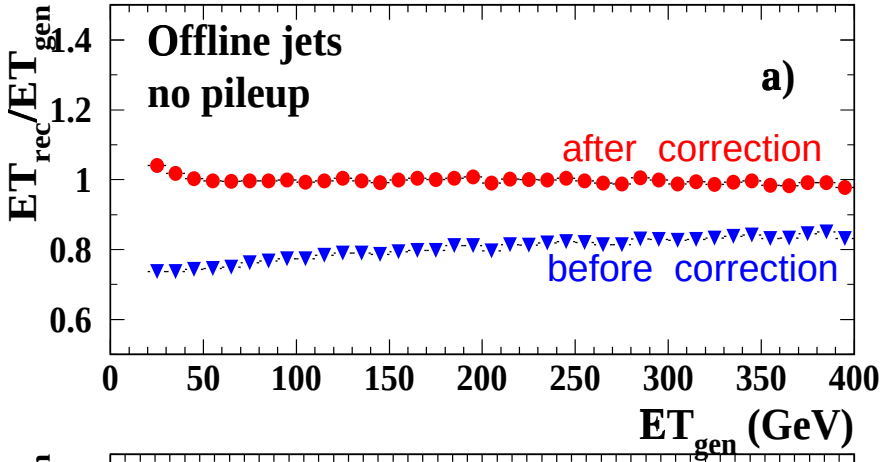


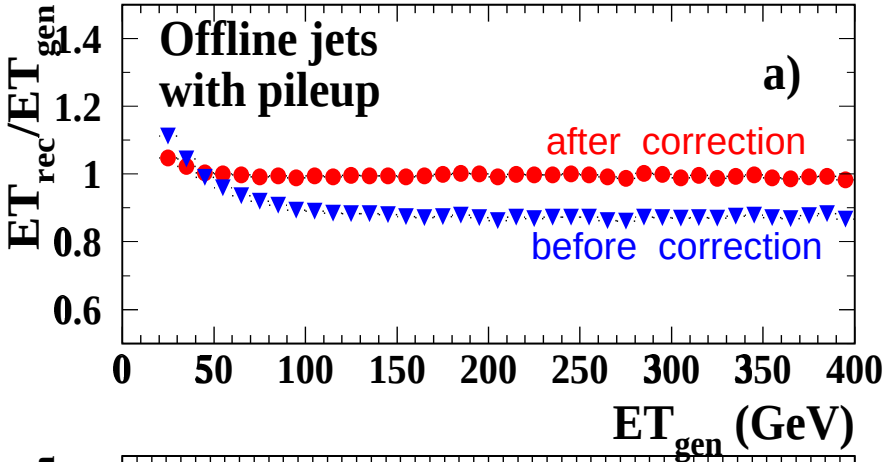


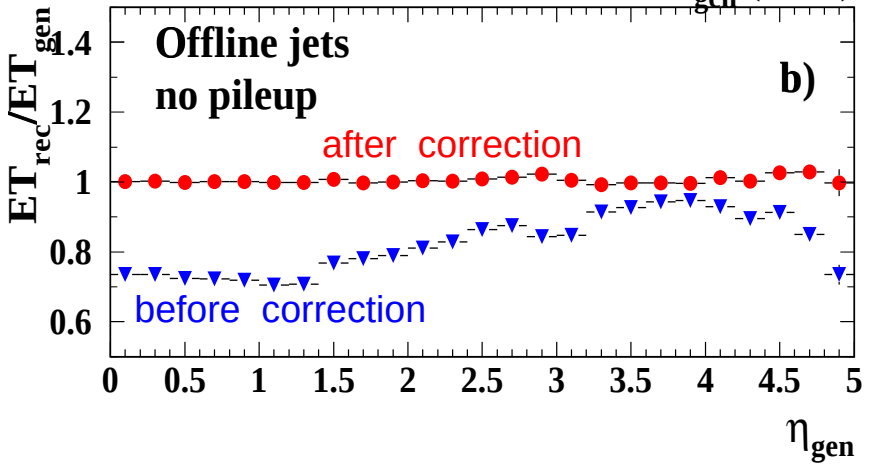
S. Arcelli et al. , CMS IN 2001/001

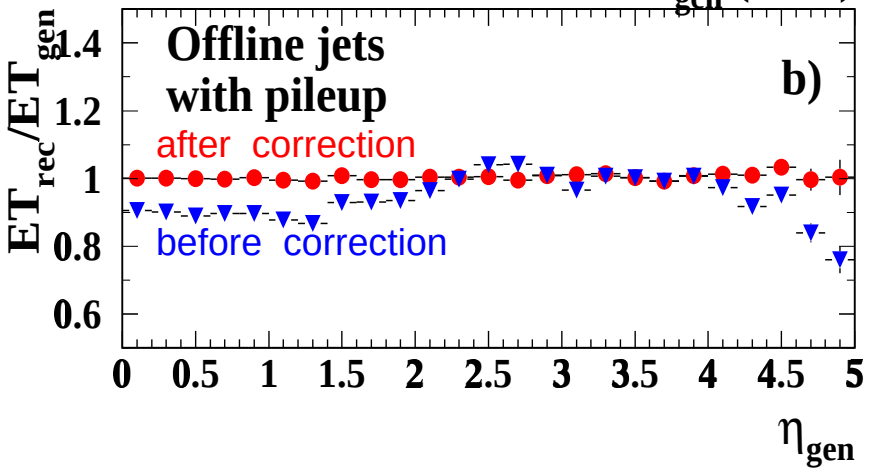
JET ENERGY CORRECTIONS (I)

- E corr = f (E rec, η) applied to L1/L2 jets with/without pile up
- Improve a single jet resolution (S. Arcelli et al., IN 2001/01)





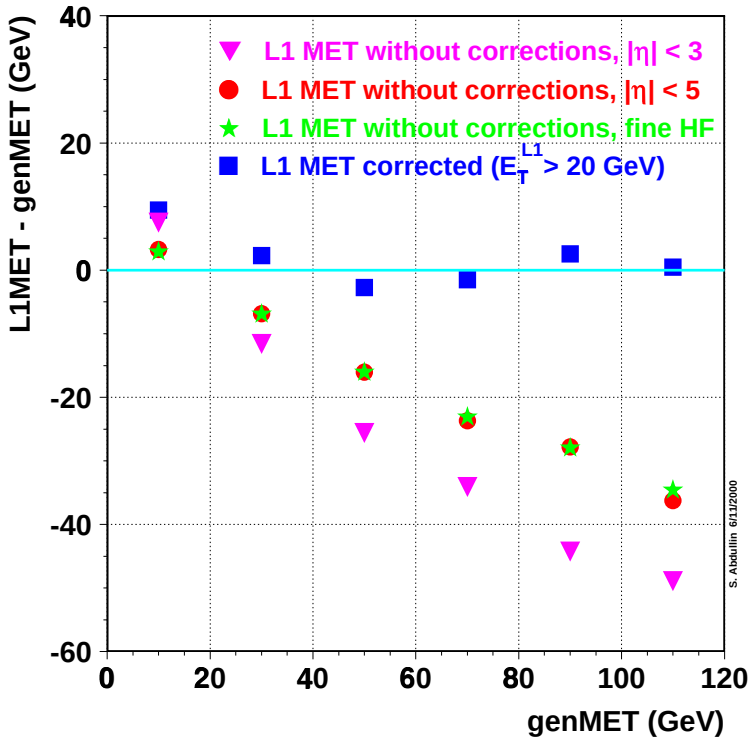




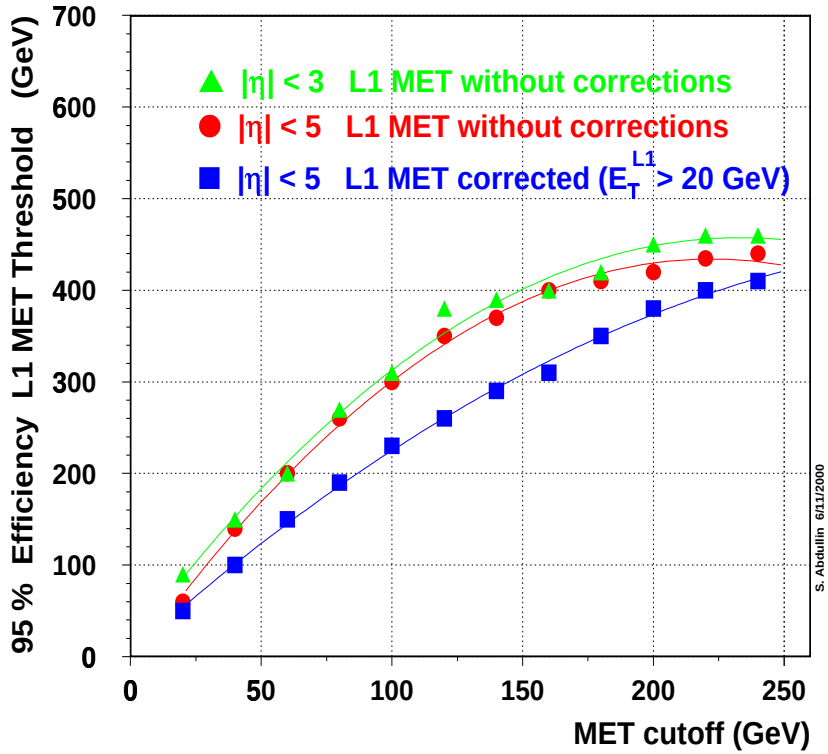
JET ENERGY CORRECTIONS (II)

- **Helpful in many aspects of jet/MET physics**
  - jj - resolution
  - MET measurement (corrected jet constituens)
- **See Sasha Nikitenko's talk for more plots ...**
- **Not a panacea !**

qqH(120 GeV) ⟶ invisible



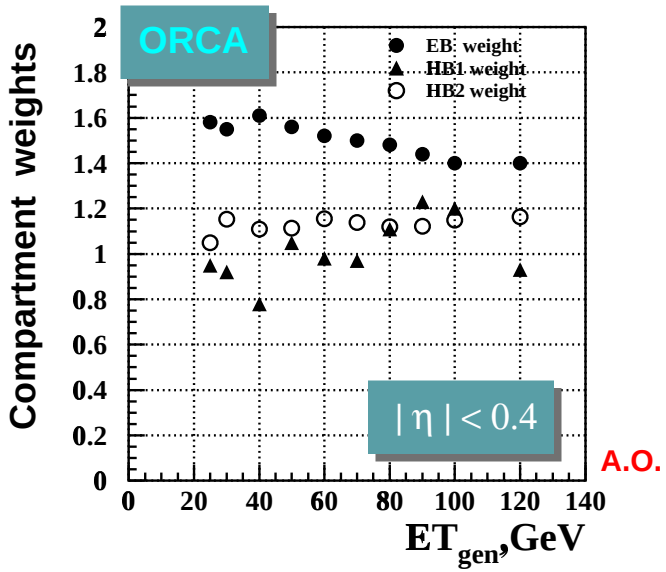
mSUGRA with M_{SUSY} ~ 500 GeV



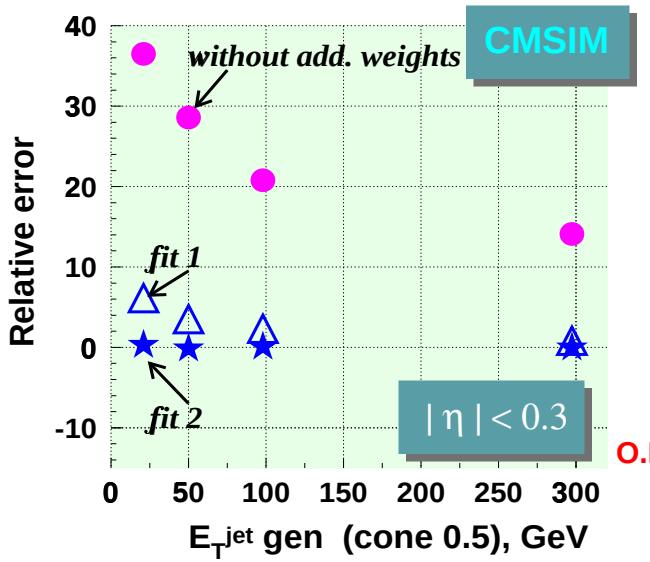
S. Abdullin, S. Eno, CMS IN 2000/060

JET ENERGY CORRECTIONS (III)

- **Another type of jet energy corrections is under consideration : optimization of the relative weights of Ecal and Hcal layers**
  - Applied "synchronously" to all constituents of the jet
  - Depend on depth layer and subdetector (HB or HE)
  - Slightly energy-dependent, but can be used as constants down to ~ 30-40 GeV
  - Intended basically for L2 jetfinding, but look applicable for L1 also.
- **Preliminary studies (O. Kodolova, I.Vardanyan, A. Oulianov) showed :**
  - In an energy-dependent way may improve resolution (a few %)
  - and set proper mean value of the jet energy
  - http://home.fnal.gov/~sceno/jpg/minutes/dec202000/olga/



A.O.



O.K and I.V.

JET ENERGY CORRECTIONS in ORCA

- **CombinedDet/JetFinders contains interface (H.-P. Wellish)**
  - WithPileupOfflineCaloCalibrator.h
  - NoPileupOfflineCaloCalibrator.h
  - WithPileupLevel1CaloCalibrator.h
  - To be tested yet
- **JetMetAnalysis/JetMetCollection (maintained by P. Hidas)**
  - HepDouble JetMetAnalysis::jetCorL2PU(HepDouble et, HepDouble eta)
  - for a while : off-line jets with pile up
- 

(who said "FORTRAN" ?)
  - still in use for Ntuples analysis
  - http://home.fnal.gov/~sceno/jpg/silviajetcorr/index.htm

JET ENERGY CORRECTIONS : PLANS

- **Update on corrections for autumn 2000 production (A. Krokhotine)**
  - **CMSIM 120 : HCAL calibrated with single pions of E_T = 30 GeV**
  - **Status :** *Coming Soon!*

- **Continue studies of "compartment weighting" technique for L1/L2 jet energy corrections**

- **HCAL electronics in ORCA is undergoing a serious "surgery"; it will certainly require update on many aspects of simulation including the jet energy corrections**

JETS/MET NTUPLE MAKER

HCAL/JET/MET Ntuple Maker

Pal Hidas

http://home.fnal.gov/~sceno/jpg/ntuple_maker/Ntuples.html

- The content of the ntuples :
  - See here the content
- How to get ready ntuples :
  - CERN Fall 2000 production
  - CERN Spring 2000 production
- How to create the executable :

ORCA_4_4_0_optimised :

Use the check-out script otherwise follow the instructions below.

ORCA_4_5_0 :

There is a check-out script which creates an `ORCA_4_5_0` tree in your `$SCRATCH` area and then builds the Ntuple maker executable named `JetMet` in `$SCRATCH/ORCA_4_5_0/bin/Linux__2.2/` or in `$SCRATCH/ORCA_4_5_0/bin/SunOS__5.6/`, depending on which architecture you use.

One needs to copy the script to one of your directories and run it. It will not destroy your existing ORCA tree, only updates it. If you want to remove your existing ORCA tree for some reason you can "source" this script with the "new_orca" parameter added (i.e. "source checkout450.csh new_orca").

- How to rebuild the executable in case of you have modified the

- **Extracts (persistent data)**
  - particle-level info
  - RecHits
  - TrigPrim
  - ...
- **Creates (via ORCA)**
  - L1 objects
  - EcalPlusHcalTower
  - jets from various objetcts
  - MET instances
  - ...
- **Analyzes**
  - jet matching with partons
  - jet matching with jets of another type
  - jet constituents rescaling
  - ...
- **Stores**
  - HBOOK Ntuples

TOWARDS PERSISTENT OBJECTS : MET

MET Classes of the HCAL/JET/MET Ntuple Maker

http://home.fnal.gov/~sceno/jpg/ntuple_maker/physics/MET_object.html

- The classes
  - BasicMET :

    It is the base class of the hierarchy with some virtual functions and the destructor. It may be superfluous, but it can be extended to be a base class of other global quantities, like event shape variables, etc.
  - SimpleMET :

    It is the general implementation class for the hierarchy of MET classes, containing the data members. It provides some simple arithmetics and an ostream operator.
  - RecMET :

    It is the templated construction interface for the hierarchy of MET classes. It has constructors specific for detector types, and nothing else. One only needs to include the header file of this class to use the whole hierarchy.
- Usage description

The myAnalysis function of JetMetAnalysis contructs new SimpleMET instances at the beginning of each event, then calls the detector analysis functions, and at the end destroys the SimpleMET instances. The pointers to them are private members of the JetMetAnalysis class, so they are available through the whole analysis.

The RecMET objects are local to the detector specific member functions of the analysis class, now renamed to JetMetAnalysis. One or two objetcs are initialized at the beginning of each function, then they are populated iterating through the detector objects (particles, jets, towers, etc.), using its SimpleMET base class addItem functions.

SimpleMET

`SimpleMET` is the general implementation class for the hierarchy of MET classes of `HepDouble` data members and member functions. It provides some simple arithmetics and an ostream operator. The `RecMET` class inherits from it.

Author

Pal Hidas

See also

`BasicMET`, `RecMET`.

Declaration

```
#include "Workspace/SimpleMET.h"
class SimpleMET : public BasicMET
```

Public Member Functions

Constructor
`SimpleMET()`

x
`HepDouble & x() const`
Return the x coordinate of MET.

y
`HepDouble & y() const`
Return the y coordinate of MET.

phi
`HepDouble & phi() const`
Return the azimuth angle of MET.

theta
`HepDouble & theta() const`
Return the polar angle of MET. For L1 global case it returns 9999.

eta
`HepDouble & eta() const`
Return the pseudorapidity of MET. For L1 global case it returns 9999.

et
`HepDouble & et() const`
Return the transverse energy of MET, i.e. MET itself. It cares for the special case of L1 global MET.

sumofet
`HepDouble & sumofet() const`
Return the scalar transverse energy, i.e. the scalar sum of MET.

So far private members are calcutated in "external" iterators using addItem () methods. It looks OK when used solely in Jets/MET Ntuple maker.

Probably class constructor has to be used for this purpose in general.

TOWARDS PERSISTENT OBJECTS : TAU

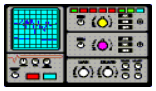
- L1 tau objects from Trigger/L1CaloTrigger (S. Dasu)
- JetMetAnalysis/HighLevelTauAnalysis (A. Nikitenko)
  - L2H2taujj class
    - L1 and L2 selection of $A \rightarrow \tau\tau \rightarrow jj$ channel
    - provides an access to L2 τ - jets for further selection with tracker
  - My_Jet class
    - contains various jet data required for the analysis
  - test/HighLevelTauTrigger.cpp
    - performs combined L1, L2 and L3 pixel (from D. Kotlinski) analysis
  - B/τ community starts to use it for regional track finder

TOWARDS PERSISTENT OBJECTS : JET

- **Jets are objects depending on various conditions/parameters**
  - JetFinders algorithm
  - Cone size or other metrics
  - Seed cut
  - E_T^{jet} cut
  - Splitting/joining criteria (not yet implemented)
  - ...
- **Sasha created mentioned earlier class (container) for his own purposes**
- **Should it exist as a persistent object**

- **More globally : we need high-level reconstructed objects**
  - MET
  - jet
  - electron
  - gamma
  - muon
  - ...

UPDATE on HCAL ELECTRONICS : PREFACE

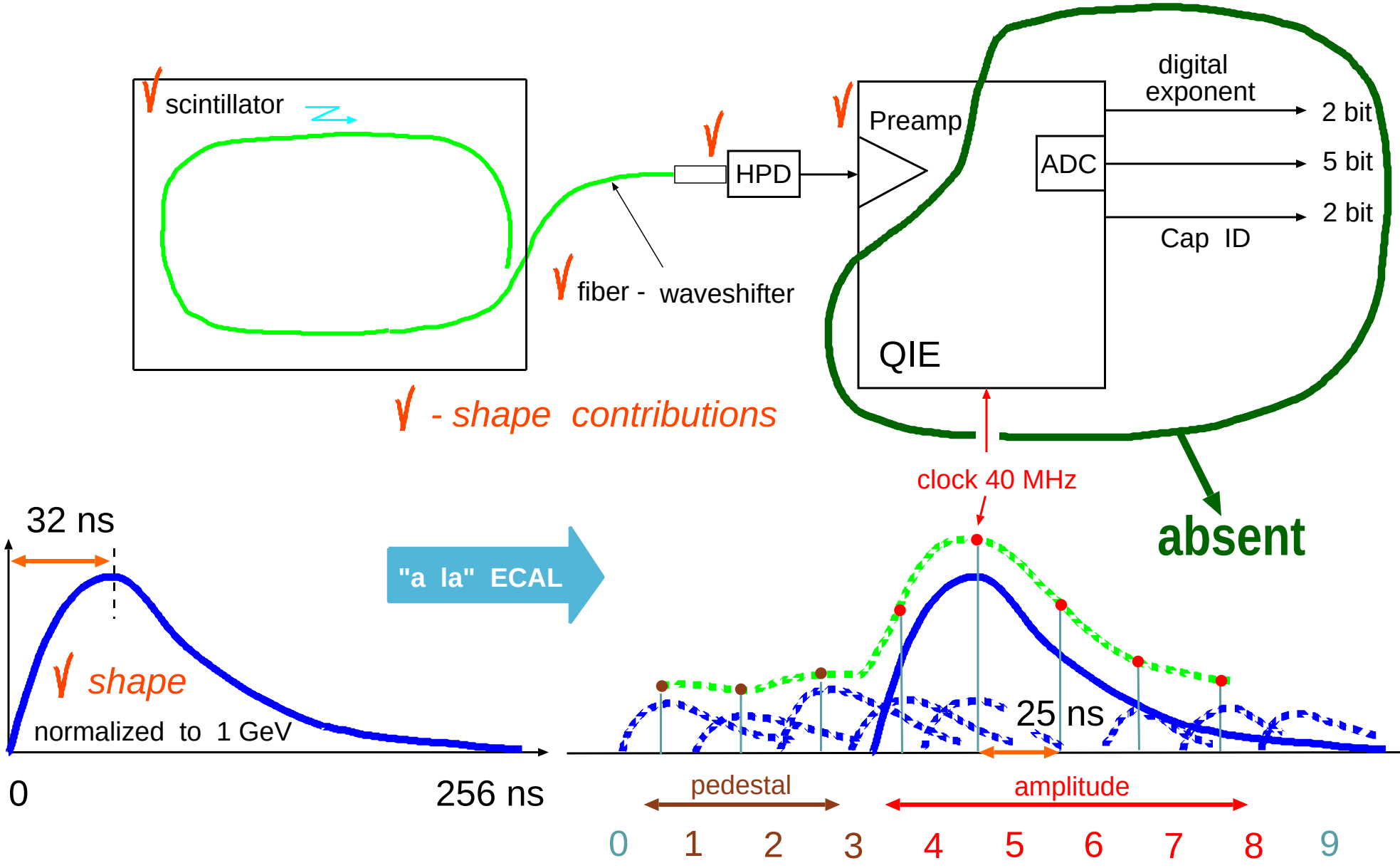


- Currently (incl. 4_5_1) an "amplitude reconstruction" using linear combination of ampitude measurements in (-3, +5) time samples is being performed ("a la" ECAL)

- After CMS week (March, 5-10) a completely different scheme of QIE integration (including ADC quantization) instead of amplitude sampling is tested

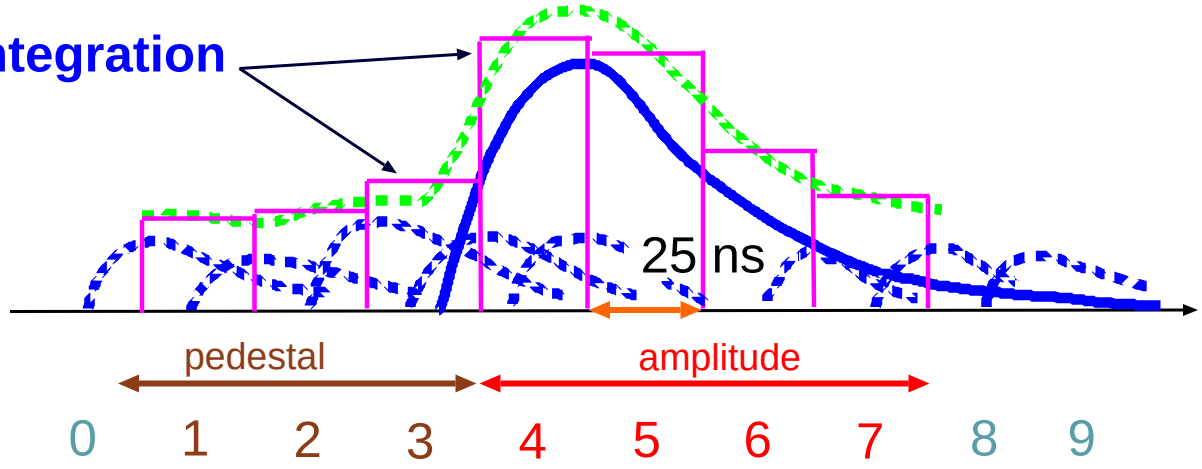
- Evererithing (noise, LSB etc.) expressed in terms of photoelectrons and requires some update (currently used 10 pe / GeV) from J.Elias et al.

HCAL ELECTRONICS : CURRENT SITUATION



HCAL ELECTRONICS : MAIN CHANGES

- **QIE integration**

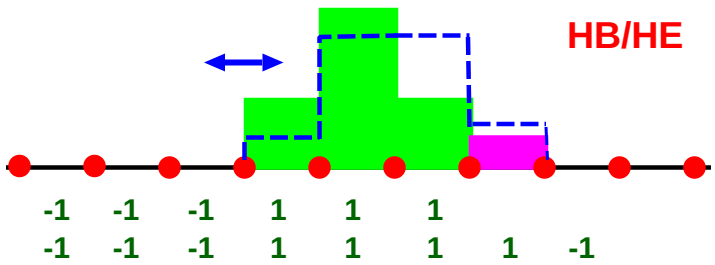


- **ADC quantization according to FADC table with 4 variable-size bin ranges.**
- **Photo statistics effect**
- **HB/HE : GEANT Hit jitter**
  - **HCAL Hit time step = 1 ns starting from CMSIM 118, not 10 as before !**
- **HF splitted from HB/HE**

HCAL ELECTRONICS : HF and HB/HE

- **Separation is not straightforward as HF/HB/HE are all in "HCAL"**
  - if (MyCell().WhichSubDetector() == "HF") is required everywhere
- **HF and HB/HE divergencies**
  - photostatistics variation : already taken into account for HF in CMSIM
  - noise : 2 pe (HB/HE) and 0.125 pe (HF)
  - shape : short HF signal (peak time ~ 3-4 ns) and longer HB/HE one (32 ns)
  - QIE integration : HF signal can be integrated within one time bucket
  - QIE sensitivity : LSB = 0.43 pe for HF and 3 pe for HB/HE
  - weights : HB/HE = (-1, -1,-1, 1, 1, 1, 1, -1) in "off-line" and
    (-1, -1, -1, 1, 1, 1) in TPG,
  - HF = (-1/3, -1/3, -1/3, 1) in "off-line" in TPG
  - time phase tuning :

    (HF doesn't have proper GEANT time)



HCAL ELECTRONICS : NOISE

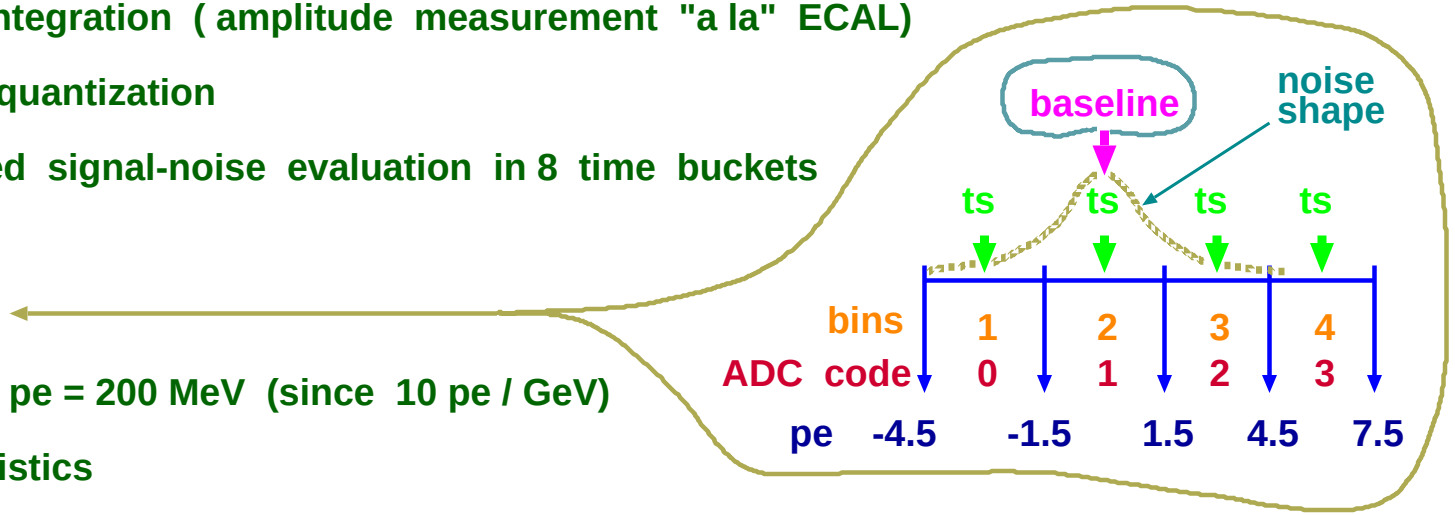
- **Noise was small (wrong ?) in ORCA** 
  - 0.0006 GeV (GEANT Hit) ~ 43, 88, and 142 MeV of reconstructed energy in HB
  - unique both for HF and HB/HE !
- **Now it increases significantly**
  - HB/HE : 2 pe ~ 200 MeV per readout (10 pe / GeV), while LSB = 3 pe !
  - HF : 0.125 pe ~ 200 - 300 MeV per readout (doesn't play a big role)
- **First observations :**
  - HB : excessive number of towers with $E_T > 1$ GeV from noise !
  - very few before !
- **Work is under way ...**

NOISE : CALCULATIONAL DETAILS (I)

OLD

- noise = 0.0006 GeV (effectively means 43, 88 and 142 MeV in HB)
- no photostatistics effect
- no QIE integration (amplitude measurement "a la" ECAL)
- no ADC quantization
- conjugated signal-noise evaluation in 8 time buckets

NEW 1

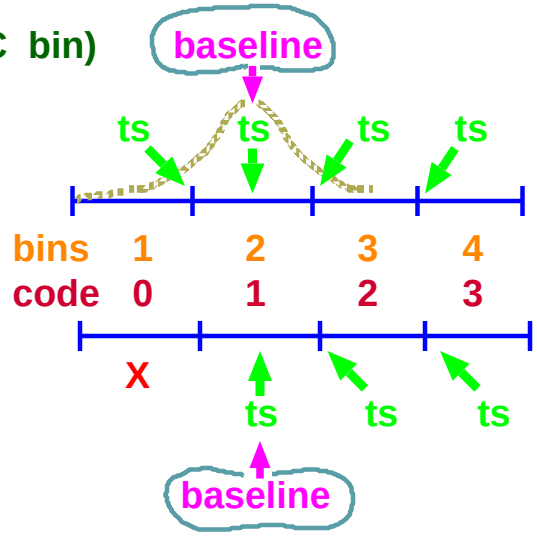


- noise = 2 pe = 200 MeV (since 10 pe / GeV)
- photostatistics
- QIE integration : code -> energy - the middle of the bin is assigned
- ADC quantization : LSB = 3 pe, baseline = 0 pe (2nd ADC bin)
- TrigPrim weights = -1.5, -1.5, -1, 1, 1 (5)
- RecHit weights = -2, -2, 1, 1, 1, 1 (6)

NOISE : CALCULATIONAL DETAILS (II)

NEW 2

- noise = 2 pe = 200 MeV (since 10 pe / GeV)
- photostatistics
- QIE integration : code -> energy - the edge closest to baseline is assigned
- ADC quantization : LSB = 3 pe, baseline = 0 pe (2nd ADC bin)
- TrigPrim weights = -1, -1, -1, 1, 1, 1 (6)
- RecHit weights = -1, -1, -1, 1, 1, 1, 1, -1 (8)



NEW 3

- noise = 2 pe = 200 MeV (since 10 pe / GeV)
- photostatistics
- QIE integration : code -> energy - the edge closest to baseline is assigned
- ADC quantization : LSB = 3 pe, baseline = 0 pe (2nd ADC bin), code = 0 - ignored
- TrigPrim weights = -1, -1, -1, 1, 1, 1 (6)
- RecHit weights = -1, -1, -1, 1, 1, 1, 1, -1 (8)

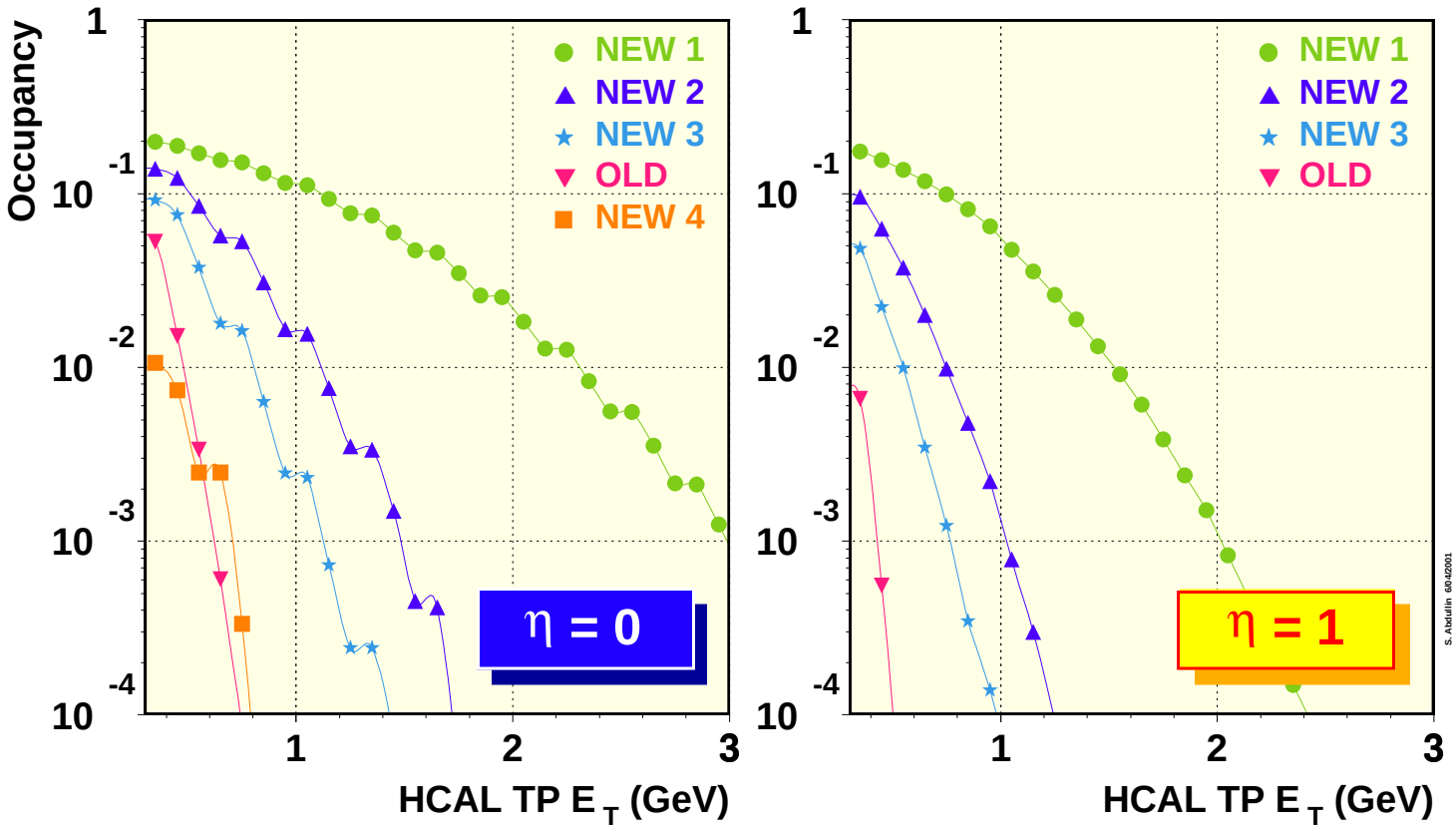
NOISE : CALCULATIONAL DETAILS (III)

bin No	FADC code	bin energy (MeV)	code → energy		
			NEW 1	NEW 2	NEW 3
1	0	-450 to -150	-300	-150	0
2	1	-150 to 150	0*	0*	0*
3	2	150 to 450	300	150	150
4	3	450 to 750	600	450	450
5	4	750 to 1050	900	750	750

baseline

* **important :** results slightly depend on the baseline variation within one ADC bin

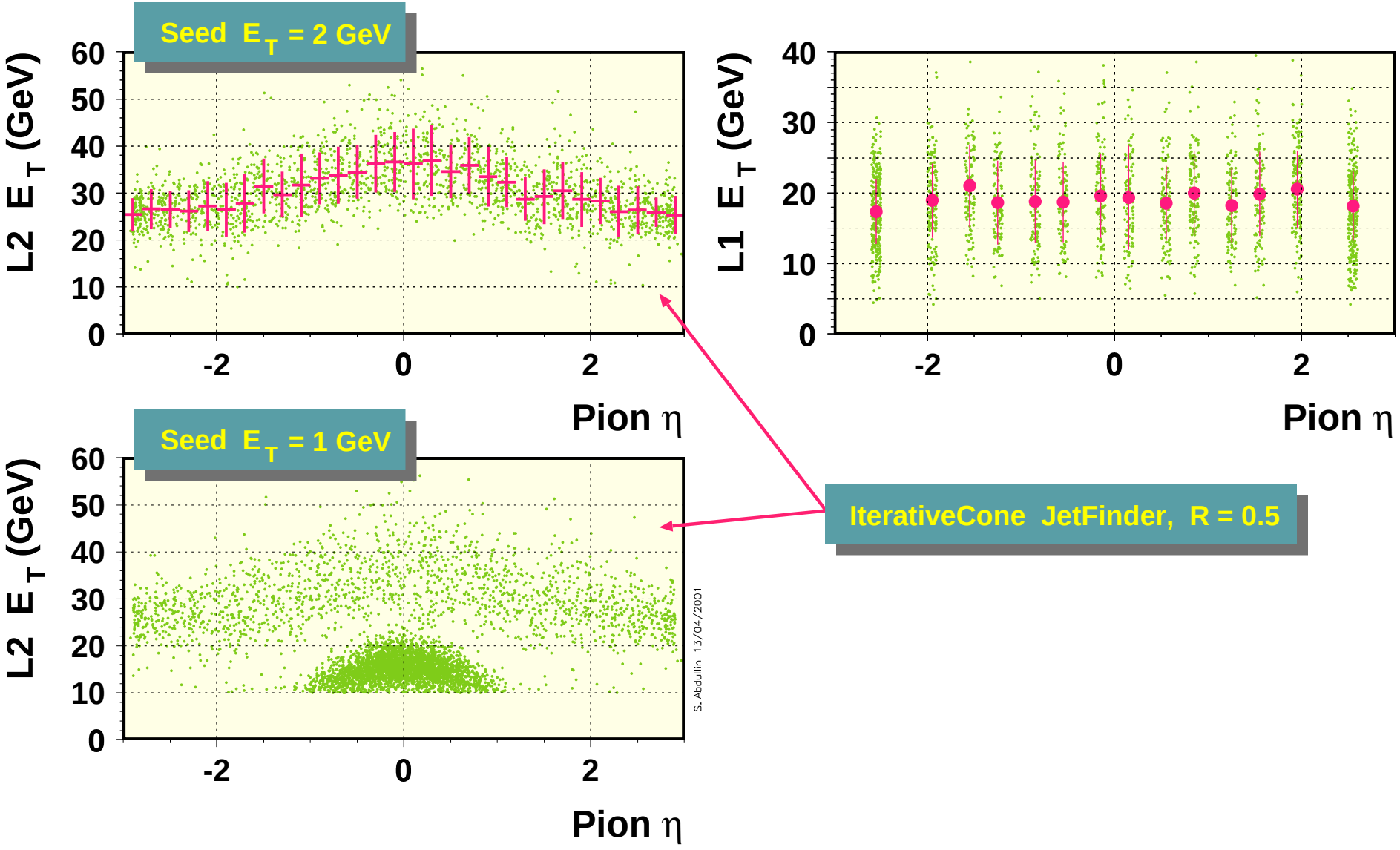
HCAL TRIGGER PRIMITIVE "OCCUPANCY" due to NOISE



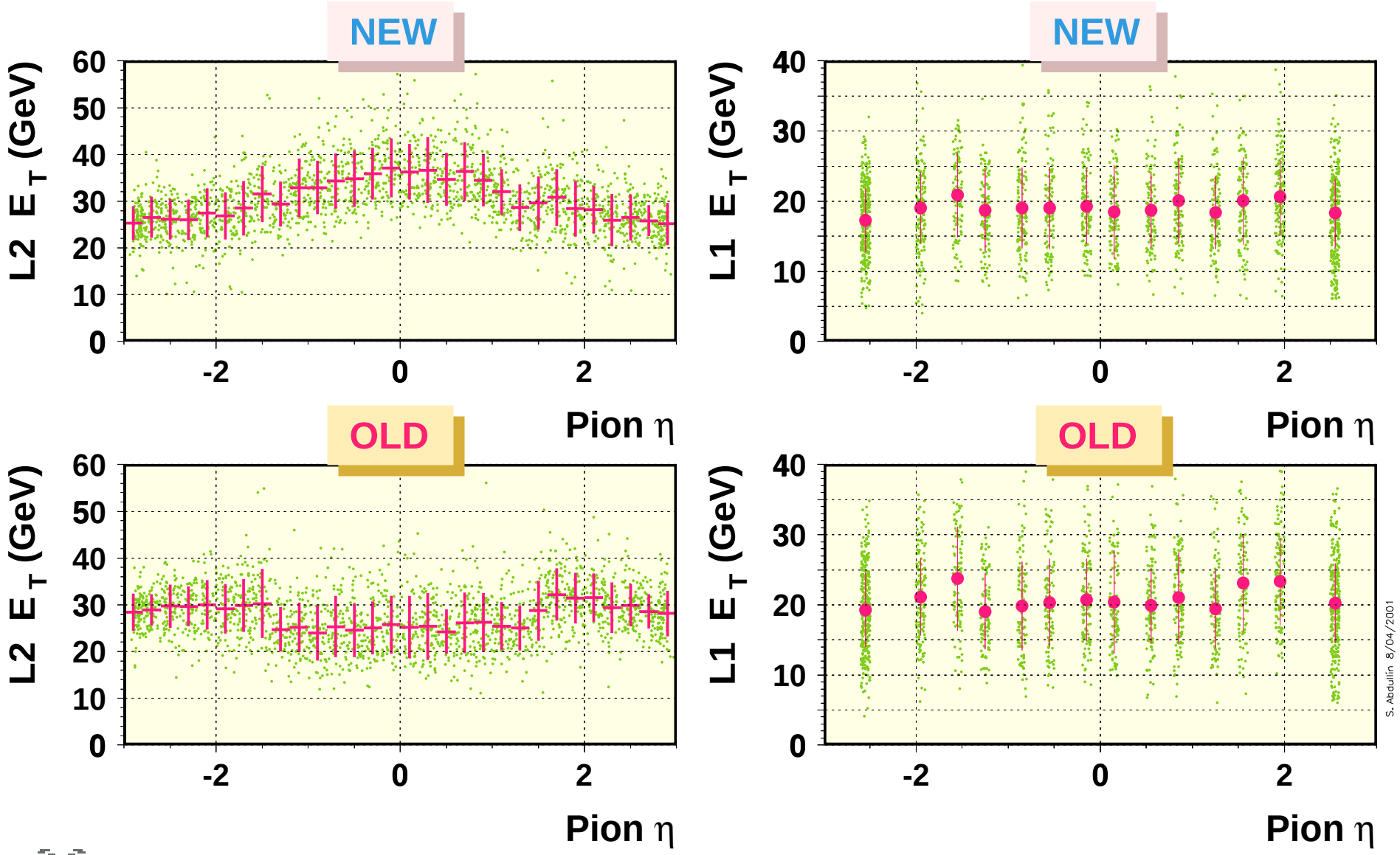
- **NEW 1 :** weights = -1.5, -1.5, 1, 1, 1, code -> energy : the middle of the bin
- **NEW 2 :** weights = -1, -1, -1, 1, 1, 1, code -> energy : the closest to the baseline
- **NEW 3 :** weights = -1, -1, -1, 1, 1, 1, code -> energy : idem + code=0 ignored
- **NEW 4 :** idem + one readout, the simplest noise filter ☜ *yesterday night*

L1, L2 JETS from SINGLE PIONS (I)

- NEW 3

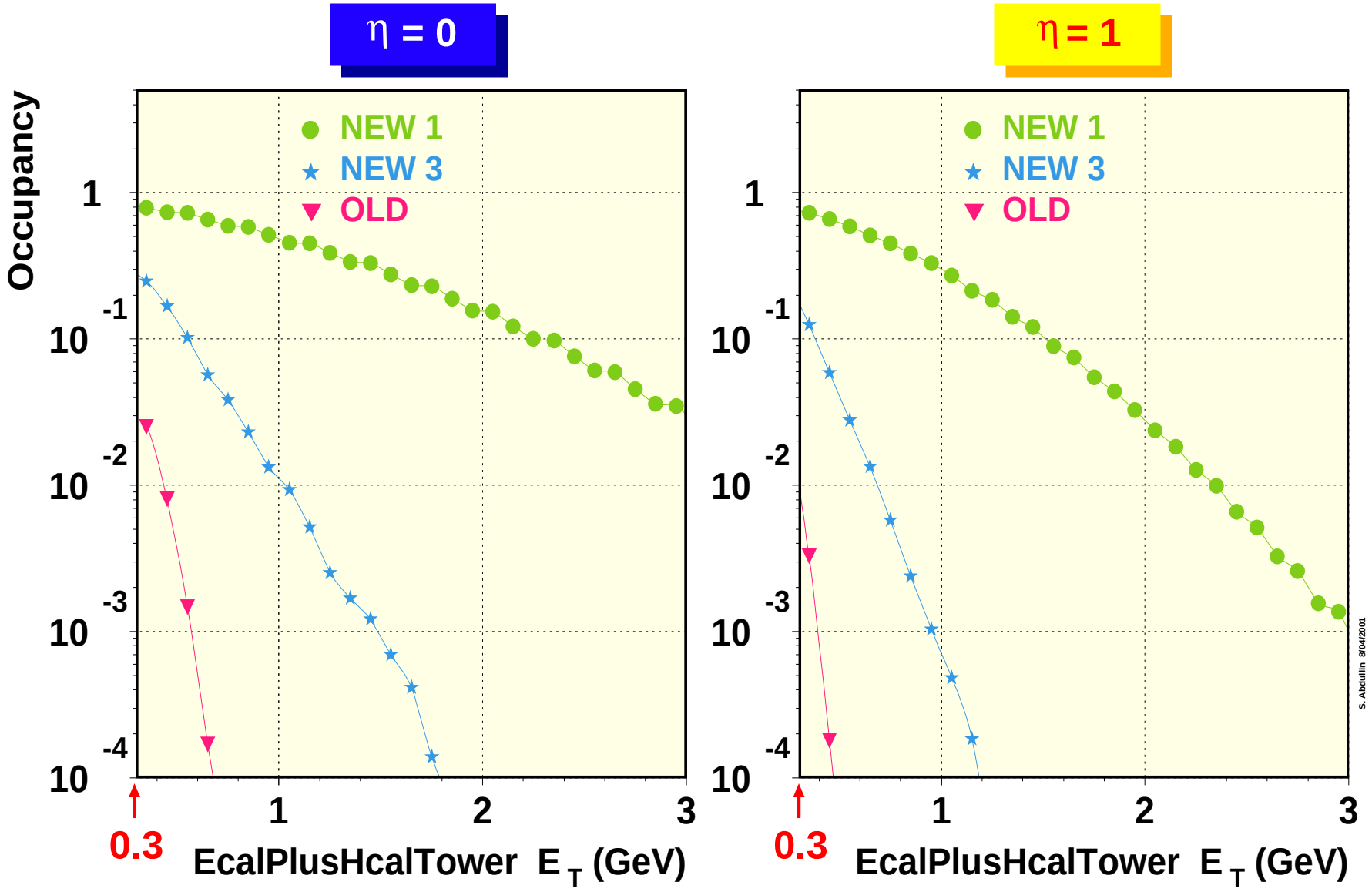


L1, L2 JETS from SINGLE PIONS (II)



CALO TOWER "OCCUPANCY" due to NOISE

- NEW 3 : RecHit reconstruction weights = -1, -1, -1, 1, 1, 1, 1, -1



PLANS

(Coming Soon!)

- Accomplish electronics update
- Test the performance of new HCAL
- Optimize algorithms of the energy reconstruction (L1 and off-line)
  - Time buckets weights
  - Zero suppression (L1)
  - Pedestal subtraction
  - Bunch crossing ID
  - Out-of-time suppresion
  - ...
- Update "global" jet energy corrections
- Develop "compartment weighting" jet energy corrections
  - To implement in L1 look-up tables
  - L2 EcalPlusHcalTowers correction
- ...